MULTIPHASE CLOCK RECOVERY USING D-TYPE PHASE DETECTOR

Cross-Reference to Related Application

This application is related to U.S. Patent Application Serial No. 09/ 752,094 entitled "SELF-CORRECTING MULTIPHASE CLOCK RECOVERY" (attorney docket number AUS9-2000-0519), filed concurrently with this application, which is hereby incorporated.

Background of the Invention

1. Field of the Invention

The present invention generally relates to electronic clock circuits, specifically to a method and system for providing clock recovery from a high-frequency data signal, and more particularly to such a method and system that has reduced power dissipation, and acceptable cycle variation (jitter).

2. Description of Related Art

Electronic circuits that provide clock signals are used in a wide assortment of devices, and particularly in computer systems. Microprocessors and other computer components, such as random access memory (RAM), device controllers and adapters, use clock signals to synchronize various high-speed operations. These computer clock circuits often use a phase-lock loop (PLL) circuit to synchronize (de-skew) an internal logic control clock with respect to an external system clock.

A typical prior art PLL circuit **1** is shown in **Figure 1** and includes a phase/frequency detector (PFD) **2**, a charge-pump **3**, a low-pass filter **4**, and a voltage-controlled

oscillator (VCO) **5**. Phase/frequency detector **2** compares two input signals, a reference signal f_{ref} (from the external system clock) and a feedback signal f_{fb}, and generates phase error signals that are a measure of the phase difference between f_{ref} and f_{fb}. The phase error signals ("UP" and "DOWN") from detector **2** are used to generate control signals by charge-pump **3** which are filtered by low-pass filter **4** and fed into the control input of voltage-controlled oscillator **5**. Voltage-controlled oscillator **5** generates a periodic signal with a frequency which is controlled by the filtered phase error signal.

The output of voltage-controlled oscillator **5** is coupled to the input f_{fb} of phase/frequency detector **2** directly or indirectly through other circuit elements such as dividers **6**, buffers (not shown) or clock distribution networks (not shown), thereby forming a feedback loop. If the frequency of the feedback signal is not equal to the frequency of the reference signal, the filtered phase error signal causes the frequency of voltage-controlled oscillator **5** to shift (upwards or downwards) toward the frequency of the reference signal, until voltage-controlled oscillator **5** finally locks onto the frequency of the reference; following frequency acquisition, phase acquisition is achieved in a similar manner. The output of voltage-controlled oscillator **5** is then used as the synchronized signal (for internal logic control).

In cases where the incoming data is a self-clocking bit stream, the comparator system may be used to extract (recover) the clock information from the data stream itself.

Clock extraction for high-speed serial links is usually accomplished using a VCO with a center frequency N for extracting a clock from a serial data stream modulated at N bits/sec. The VCO may provide multiple phases for oversampling, as discussed in the article by Yang, et al.,"A 0.5 um CMOS 4.0 Gbit/s Serial Link Transceiver with Data Recovery Using Oversampling," IEEE JSSC Vol 33, No. 5 (May 1998), or more commonly offer just a single phase for direct detection. An example of a single phase D-type phase detector technique used for non-return-to-zero (NRZ) data is disclosed in the article by Boerstler, "Dynamic Behavior of a Phase-Locked Loop Using D-Type Phase Detector and Nonlinear Voltage-Controlled Oscillator", IBM Technical Report TR 21.1428 (March 21, 1991).

For high-bandwidth applications such as in packet switches, maximizing both frequency and density simultaneously is desired, but this causes power dissipation and/or power density to be a significant problem. CMOS technology limitations can also limit the speed at which clock recovery can be accomplished, and operating the clocks at one-half the baud rate has been reported, as in Ewen et al., "Single-Chip 1062 Mbaud CMOS Transceiver for Serial Data Communication," ISSCC Digest, Vol 38, pp. 32-33 (Feb. 1995).

In light of the foregoing, it would be desirable to devise an improved method for recovering a high-speed clock from a data signal. It would be further advantageous if the method were to result in reduced power dissipation, while still ensuring acceptable levels of jitter.

Summary of the Invention

It is therefore one object of the present invention to provide an improved clock circuit, such as may be used with a microprocessor or other high-performance computer components.

It is another object of the present invention to provide such a clock circuit which is able to extract a clock signal from a data stream having a high data rate.

It is yet another object of the present invention to provide a method of recovering clock signal from a data stream which results in decreased power dissipation as compared to the prior art.

The foregoing objects are achieved in a method of extracting a clock signal from a data stream, generally comprising the steps of generating a plurality of multiphase clock signals, selecting one of the multiphase signals based on a plurality of synchronization states identifying which of the multiphase clock signals is most closely aligned with the data stream, and sampling the data stream using the selected one of the multiphase signals to produce a retimed data signal. The multiphase clock signals may be subharmonics of the data stream. The selecting step may include the determination of whether the multiphase clock signals are either early or late with respect to the data stream, particularly using D-type flip-flops. The synchronization states are used to define which of the rising edges of the multiphase clock signals is most closely aligned with an edge of the data stream. A multiphase voltage-controlled oscillator may be used to provide the multiphase clock signals. An error signal is created using

the multiphase clock signals and the data stream which is applied to a charge pump, and the multiphase clock signals are corrected using a control voltage output of the charge pump.

The above as well as additional objectives, features, and advantages of the present invention will become apparent in the following detailed written description.

Brief Description of the Drawings

The novel features believed characteristic of the invention are set forth in the appended claims. The invention itself, however, as well as a preferred mode of use, further objectives, and advantages thereof, will best be understood by reference to the following detailed description of an illustrative embodiment when read in conjunction with the accompanying drawings, wherein:

Figure 1 is a block diagram illustrating a conventional phase-lock loop;

Figure 2 is timing diagram illustrating synchronization and retiming in one implementation of the present invention using a multiphase voltage-controlled oscillator (VCO) and a D-Type phase-frequency detector;

Figure 3 is a table illustrating synchronization and retiming state identification information used according to the present invention;

Figure 4 is a table illustrating how the timing correction (early/late) is determined in accordance one implementation of with the present invention which uses a D-type phase detector;

Figure 5 is a timing diagram illustrating the multiphase clock signals and synchronization states with clock and data aligned, for the D-type phase detector implementation;

Figure 6 is a timing diagram illustrating multiphase clock signals and synchronization states where the clock lags the data signal, for the D-type phase detector implementation;

Figure 7 is a timing diagram illustrating the multiphase clock signals and synchronization states when the clock leads the data signal, for the D-type phase detector implementation;

Figure 8 is a block diagram illustrating a multiphase phase-lock loop using a D-type phase detector in accordance with one embodiment of the present invention;

Figure 9 is a block diagram of the multiphase phase detector used in the phase-lock loop of **Figure 8**;

Figure 10 is a block diagram of the sampled clock unit used in the multiphase phase detector of **Figure 9**;

Figure 11 is a block diagram illustrating a latch array used to sample the clock signals, in the sampled clock unit of **Figure 10**;

Figure 12 is block diagram of a switchport (multiplexor) used by the sampled clock unit of **Figure 10**;

Figure 13 is a block diagram illustrating the retime latch of **Figure 9**;

Figure 14 is a block diagram of another multiphase phase-lock loop using self-correcting phase detector, in accordance with another embodiment of the present invention;

Figure 15 is a block diagram illustrating the multiphase phase detector used in the phase-lock loop of **Figure 14**;

Figure 16 is a block diagram of a self-correcting unit used by the multiphase phase detector of **Figure 15**;

Figure 17 is a block diagram of a sampled data unit used in the self-correcting unit of **Figure 16**;

Figure 18 is a block diagram of an XOR gate array used in the self-correcting unit of **Figure 16**;

Figure 19 is a block diagram of the multiplexor used by the multiphase phase detector of **Figure 15**;

Figure 20 is a timing diagram illustrating the various signals associated with a "retime state 1" wherein the clock and data are aligned, for the self-correcting implementation of the present invention;

Figure 21 is a timing diagram illustrating the various signals associated with a "retime state 2" wherein the clock and data are aligned, for the self-correcting implementation of the present invention;

Figure 22 is a timing diagram illustrating the various signals associated with a "retime state 3" wherein the clock and data are aligned, for the self-correcting implementation of the present invention;

Figure 23 is a timing diagram illustrating the various signals associated with a "retime state 4" wherein the clock and data are aligned, for the self-correcting implementation of the present invention;

Figure 24 is a timing diagram illustrating the various signals associated with a "retime state 5" wherein the clock and data are aligned, for the self-correcting implementation of the present invention;

Figure 25 is a timing diagram illustrating the multiphase clock signals and error correction signals wherein the clock and data are aligned, for the self-correcting implementation of the present invention;

Figure 26 is a timing diagram illustrating the multiphase clock signals and error correction signals wherein the clock and lags the data (for retime state 1), for the self-correcting implementation of the present invention; and

Figure 27 is a timing diagram illustrating a multiphase clock signals and error correction signals where the clock leads the data (for retime state 2), for the self-correcting implementation of the present invention.

Description of an Illustrative Embodiment

The present invention provides an improved method and system for extracting a clock signal from a serial data stream. In one embodiment, the invention uses a multiple-phase subharmonic clock from a non-return-to-zero (NRZ) stream to retime and deserialize the data. In particular, the implementations described below use a 5-phase one gigahertz (GHz) voltage-controlled oscillator (VCO), and either a 5-phase D-type phase detector or a 5-phase self-correcting phase detector configured in a phase-lock loop (PLL), to recover the clock and retime a 5 gigabit per second NRZ data stream. Those skilled in the art will appreciate, however, that the specific implementations disclosed herein are not to be construed in a limiting sense. For example, the invention could be used for clock reduction factors other than 5, e.g., 7, 9, 11, etc., for single-rail rings oscillators, or 4, 6, 8, etc., for differential ring oscillators, or 4 for quadrature oscillators.

With reference now to the figures, and in particular with reference to **Figure 2**, there is depicted a timing diagram for the 5 Gb/s NRZ data stream implementation of the invention. Five clock phases are available from a multiphase VCO operating at one GHz. An appropriate multiphase VCO is disclosed in copending U.S. Patent Application Serial No. __/_____ filed on November 30, 2000, and entitled "A HIGH-FREQUENCY LOW-VOLTAGE MULTIPHASE VOLTAGE-CONTROLLED OSCILLATOR," which is hereby incorporated. The clock phases are separated equally and duty cycle is 50% for each phase. Input data has timing jitter on all edges which has a distribution around the mean time values T_1, T_2, ..., T_n.

Five synchronization states (SS_1-SS_5) are shown in **Figure 2**, defining which of the rising edges of the five clock phases is most closely aligned with the edges (positive or negative) of data signal D at any given point in time. SS_1 is asserted at time T_1 since the first phase signal $\emptyset_1$ (at **A**) is in alignment, and is deasserted at time T_2 since second phase signal $\emptyset_2$ (at **B**) becomes aligned and SS_2 is asserted. SS_2 stays asserted during the period from time T_2 to time T_3, when no transitions in the data are present, and deasserts when SS_4 is asserted due to the fact that the fourth phase signal $\emptyset_4$ (at **D**) is aligned with time T_3. Some synchronization states might not be asserted for brief periods of time due to data transitions and the currently established timing relationships (e.g., SS_5 in **Figure 2**). Data is retimed (sampled in the middle of the "eye" for optimum bit error rate (BER)) by choosing the clock phase appropriate for the current synchronization state ("SynchState"). Data is sampled at time T_7 by $\emptyset_4$ (at **L**) since the current SynchState is 1, i.e., the condition of SynchState=1 defines RetimeState=4 (RS_4). Similarly, SS_2, SS_3, SS_4, and SS_5 define the retime states RS_5, RS_1, RS_2, and RS_3, respectively.

Figure 3 shows a table which is used to determine the current SynchState and RetimeState from serial data sampling the clocks. Transitions in the data (both positive and negative) are used to sample all five phases of the clock and are held as binary values Q_1-Q_5 for $\emptyset_1$-$\emptyset_5$, respectively. The "X" values in the table of **Figure 3** are examined to generate the error signal for the PLL, according to the second table shown in **Figure 4** (for the D-type detector implementation of the present invention). For example, if

SynchState=1, then a zero value for Q_1 indicates that the clock is late ("Clock Late wrt Data" value of 1), while a one value for Q_1 indicates that the clock is early ("Clock Late wrt Data" value of 0). The timing of the error signals is shown in **Figure 2**, where the clocks are determined to be either early or late with respect to the data (for the D-type detector implementation of the present invention). Data sampling the clocks is illustrated in **Figure 2** at time T_6, where the sampled values of $Ø_1$-$Ø_5$ are latched to create values Q_1-Q_5 as shown in the timing diagram. The values of Q_1-Q_5 at T_6 correspond to SynchState=4 and RetimeState=2, shown in **Figure 2** becoming asserted at T_6, while the previous SynchState and RetimeState become deasserted (SS_3/RS_1).

The clocks sample the data signal to retime the data. **Figure 2** illustrates how the negative transition of the appropriate clock phase is used to sample the data in the center of the bit interval. For example, during the period from T_4 to T_5, SynchState=2 and RetimeState=5 are asserted; therefore the negative edge of $Ø_5$ samples the data at T_{10}, resulting in the retimed data signal (Retimed D) changing to a low state. In general, for RetimeState=n (n=1...5), a negative transition of $Ø_n$ samples D, and a positive transition of $[1+\mathrm{mod}_5(Ø_{n+2})]$ samples SD_n (the sampled data from $Ø_n$): a negative transition of $Ø_1$ samples D, positive transition of $Ø_4$ samples SD_1; a negative transition of $Ø_2$ samples D, positive transition of $Ø_5$ samples SD_2; a negative transition of $Ø_3$ samples D, positive transition of $Ø_1$ samples SD_3; a negative transition of $Ø_4$ samples D, positive transition of Ø2 samples SD4; and a negative transition of $Ø_5$ samples D, positive transition of $Ø_3$ samples SD_5.

Figure 5 shows a timing example for the condition wherein the clocks are generally aligned with the data, for the D-type detector implementation of the present invention. An error signal is generated which becomes asserted at time T_1 due to $\emptyset_1$ being late with respect to the rising edge of the data D for SS_1 asserted. The error remains high at time T_2 since $\emptyset_2$ is late with respect to the negative edge of D for SS_2 asserted. At time T_3 SynchState= 4, $\emptyset_4$ leads D, and the error signal deasserts. At time T_4 (SS_2) $\emptyset_2$ leads D and the error stays low, at time T_5 (SS_3) $\emptyset_3$ lags D and error asserts, and at time T_6 (SS_4) $\emptyset_4$ leads D and error deasserts. Since the phase detector has the characteristics of a D-type flip-flop, it responds ideally to infinitesimal phase differences between the clock and the data. When the PLL is in steady-state, the average value of the error signal is 50% of the logic swing for random input data. The error signal is integrated by a charge pump or filter circuit to produce the control voltage (V_c) for the VCO (discussed further below). The control voltage causes the VCO frequency to increase (V_c increasing) or decrease (V_c decreasing) to correct for the error.

Figure 6 shows the timing for the clock lagging the data. The average value of the error signal rapidly increases for this condition, causing the VCO frequency to increase rapidly. Similarly, **Figure 7** shows a corresponding rapid decrease in the average value of the error signal for the clock leading the data, forcing the frequency of the VCO rapidly lower. Very rapid PLL acquisition is a consequence of this phenomena, but increased jitter caused by overcorrection during missing transitions may also occur.

Figure 8 is a block diagram depicting a multiphase PLL **10** which implements the foregoing embodiment of the invention. An NRZ data stream is presented at the input to PLL **10**. A 5-phase VCO **12** operating at one GHz is connected to a D-type multiphase phase detector **14** which includes a retiming function. D-type multiphase phase detector **14** thus receives each of the signals $\emptyset_1$-$\emptyset_5$, as well as the data signal, and outputs a late clock error signal to a conventional charge-pump **16**. Charge-pump **16** creates a control (feedback) voltage for VCO **12** from the error signal.

Figure 9 is another block diagram depicting the high-level organization of D-type multiphase phase detector **14**. D-type multiphase phase detector **14** is comprised of a sampled clock unit **18**, a retime latch **20**, synchronization logic **22**, and retiming logic **24**. Sampled clock unit **18** receives the data and five clock signals, and uses both edges of the data to sample the clocks. Retime latch **20** receives these inputs as well, and uses the clocks to sample the data to create retimed data. Synchronization logic **22** generates the SynchStates according to the table of **Figure 3**, and provides the late clock signal. Retiming logic **24** generates the RetimeStates according to the table of **Figure 4**, and provides those states to retime latch **20**.

Figure 10 illustrates exemplary components of sampled clock unit **18**. Two sets of latch arrays **26** sample the clocks from the data signal to provide the latched values (a suitable latch configuration is shown in **Figure 11**). A

switchport **28** (essentially a multiplexor, see **Figure 12**) selects from among the inputs provided by the flip-flop arrays to output the appropriate data samples P_1-P_5, which are then provided to synchronization logic **22** and retiming logic **24** (the "Q" values in **Figures 3** and **4**).

Figure 13 illustrates exemplary details of how retime latch **20** selects the appropriate phase for capturing the data. The phase is selected by combining respective pairs of retime state signals and inverted phase signals using AND gates **30**, and further combining the outputs of AND gates **30** using a 5-input OR gate **32**. A D-type flip-flop **34** latches the output of OR gate **32** using the original data signal, to produce the retimed data signal.

Simulations of the foregoing design for PLL **10** demonstrate that phase acquisition time is more than an order of magnitude faster than that for alternate PLL designs. Operating the PLL at lower speeds saves power, allowing higher densities, and makes less-advanced technologies viable alternatives. The use of a D-type phase detector offers not only significantly reduced acquisition times, but also lower complexity, and less power dissipation. D-type phase detectors may require aided frequency acquisition or reduced VCO range to avoid lock at undesired harmonics of the serial data, and may have higher steady-state jitter for low data transition density. Other devices may be used to provide the on/off functionality of the early/late detection, such as a saturating amplifier.

In the foregoing design, the retiming state is used to select a signal that is indicative of whether the

appropriate multiphase clock is early or late with respect to the data signal. In an alternative embodiment, the retiming state is used to sample the data to determine the coarse phase error, and to detect the transition density for further refinement of the phase error determination. This alternative design of a PLL **40** constructed in accordance with the present invention is illustrated in **Figure 14**. PLL **40** again uses a 5-phase VCO **12** operating at one GHz, but VCO **12** is now connected to a 5-phase self-correcting phase detector **42** which includes a retiming function, as explained further below. A conventional charge-pump **16** is again used to create a control voltage for VCO **12**, from the error signals UP and DN.

Figure 15 depicts the high-level organization of self-correcting phase detector **42**. Self-correcting phase detector **42** includes a self-correcting unit **44** which uses the clocks to sample data, a multiphase phase-detector **14** which uses data to sample the clock phases and generate SynchStates and RetimeStates, and a multiplexor **48**. The details of multiphase phase detector **14** may be understood with reference to **Figures 9-13**. Self-correcting unit **44** provides 5 sets of UP/DN signals, one of which is selected by multiplexor **48** to pass on to charge pump **16**, as explained further below.

Self-correcting unit **44** is detailed in **Figure 16**, and includes a sampled data unit **50**, and an XOR gate array **52**. Sampled data unit **50** receives the data signal and the clock signals as inputs, and has 10 outputs which are further

illustrated in **Figure 17**. Sampled data unit **50** provides the clock sampling of the data function which provides the signals SD_n and SD_{n1} as described earlier (e.g., negative transition of $\emptyset_2$ samples D, positive transition of $\emptyset_5$ samples SD_2). The data is sampled by providing the inverted multiphase clock signals to a first set of five flip-flops **54**, respectively, and providing non-inverted clock signals to a second set of five flip-flops **56**, respectively. Each of the flip-flops also receives either a data signal D or a sampeld data signal SD_n.

The outputs of sampled data unit **50** are fed as inputs to XOR gate array **52**, which is further depicted in **Figure 18**. XOR gate array **52** uses these inputs to provide a set of 10 signals referred to herein as X_n and X_{n1}, which are further explained below. The first five SD_n values (from flip-flops **54**) are respectively combined with the data signal using five XOR gates **58**, to result in signals X_n. These first five SD_n values are also respectively combined with the second set of five SD_{n1} values (from flip-flops **56**) using five more XOR gates **60**, to result in signals X_{n1}. The X_n and X_{n1} signals are provided to multiplexor **48**, which selects UP_n and DN_n in accordance with the current RetimeState RS_n. Multiplexor **48** is shown in **Figure 19**. Each of the retime state signals is fed to two different AND gates **62** and **64**. Each AND gate **62** and **64** also receives a single one of the X_n and X_{n1} signals. The outputs of AND gates **62** are combined by a 5-input OR gate **66**, and the outputs of AND gates **64** are combined by another 5-input OR

gate **68**, to create the UP and DN signals for use by charge-pump **16**.

Figure 20 shows an example of the timing for RetimeState=1 for PLL **40** for the steady state (i.e., clock and data aligned) conditions. The data D is sampled by the negative transitions of $Ø_1$ and is held as SD_1. SD_1 is sampled by the next positive transition of $Ø_4$ and is held as SD_{11}. An XOR gate compares D with SD_1 to create X_1 and another XOR gate compares SD_1 and SD_{11} to create X_{11}. During the time when RS_1 is asserted, a change in the state of D from the state of SD_1 causes X_1 to become asserted until the next negative transition of $Ø_1$ causes SD_1 and D to be equal again. If the negative transition of $Ø_1$ occurs in the center of the baud interval, then X_1 will have an average value of 50%. If the data leads the clock the average value will increase; similarly, the average value will decrease for the data lagging the clock. Also during the time when RS_1 is asserted (i.e., RetimeState=1), a change in the state of D from the state of SD_1 causes SD_{11} to change state on the subsequent positive transition of $Ø_4$, asserting X_{11} for exactly one-half of the baud interval/clock phase. In this manner a reference pulse from X_{11} is provided, against which the error pulse from X_1 can be compared. Finally, RS_1 is AND-ed with X_1 and X_{11} to create the UP_1 and DN_1 signals, respectively. For the steady state case, the areas under UP_1 and DN_1 are equal; a capacitor charged/discharged from UP_1/DN_1 through a charge pump will not register a voltage change over the duration of RetimeState1.

Figures 21, 22, 23, and **24** similarly show steady state timing relationships for RetimeStates=2, =3, =4, and =5, respectively. It will be noted that in those figures, the area under UP_n and DN_n are equal for the steady state case. **Figure 25** shows the timing involved with the various UP_n and DN_n signals, and the control voltage after the charge pump. The overall change in control voltage is zero for the steady state case when data and clock are aligned. **Figure 26** shows an example of the timing for RetimeState=1 for the clock lagging the data. Since $Ø_1$ should normally have its negative transition in the center of the eye when RS_1 is asserted, the average value of X_1 increases and UP_1 is affected similarly (see **Figure 20**). The average values of X_{11} and DN_1 remain at 50%. A positive change in control voltage will result from such a phase relationship as shown by "V_c effective" in **Figure 26**. **Figure 27** shows an example of the timing for RetimeState=2 for the clock leading the data. Note the corresponding decrease in average value for X_2 and UP_2 when RS_2 is asserted compared to **Figure 21**. Again, the average value of X_{21} and DN_2 remain at 50%. A negative change in control voltage will result from such a phase relationship as shown by "V_c effective" in **Figure 27**.

Although the invention has been described with reference to specific embodiments, this description is not meant to be construed in a limiting sense. Various modifications of the disclosed embodiments, as well as alternative embodiments of the invention, will become apparent to persons skilled in the art upon reference to the description of the invention. It is therefore contemplated that such modifications can be made without departing from

the spirit or scope of the present invention as defined in the appended claims.